



Super Computación y Cálculo Científico  
Universidad Industrial de Santander



---

# SLURM

Simple Linux Utility for Resource Management

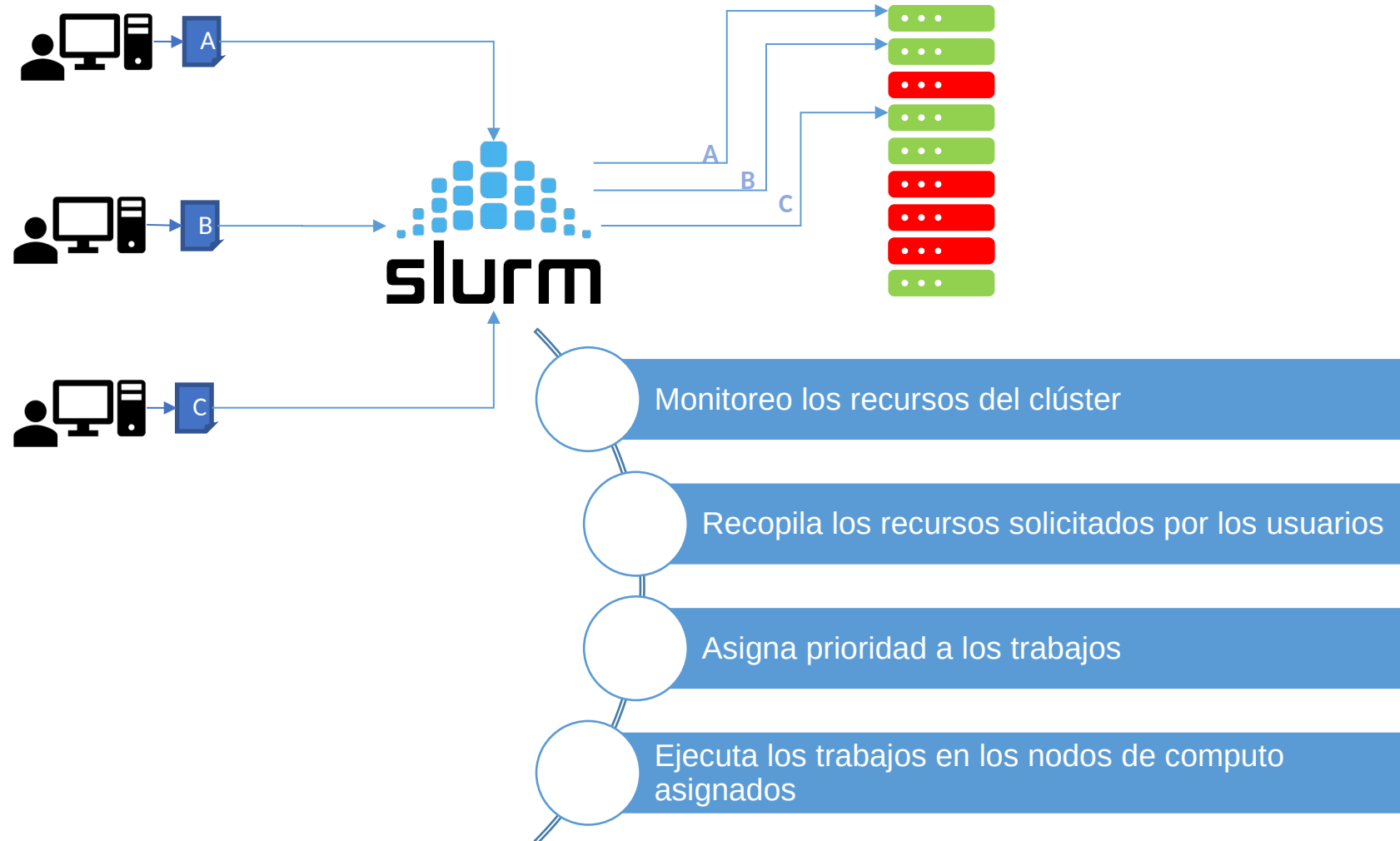
---

## Use on GUANE-1






---

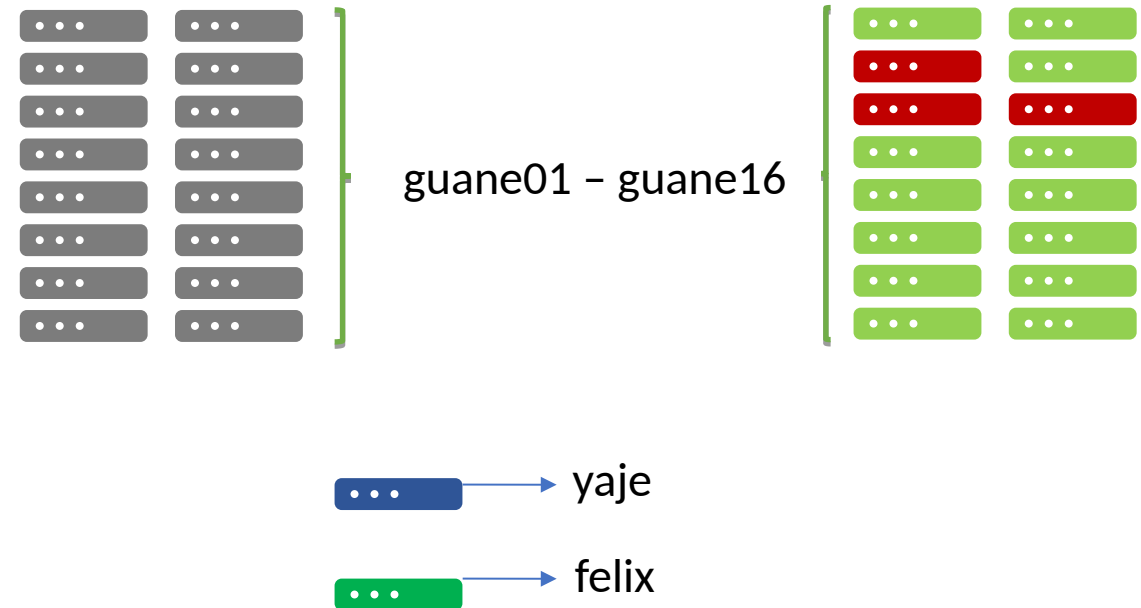
# ¿QUÉ ES SLURM?

SLURM es un software de código abierto para la administración de trabajos y gestión de clústeres Linux.



Los nodos de computación se agrupan en conjuntos lógicos llamados particiones que dependen de sus características hardware o función:

	PARTITION
	normal*
	guane_16_cores
	guane_24_cores
	Viz
	deepL



# CONEXIÓN A GUANE

SSH `username@ip-address` or `hostname`



`ssh user_name@167.249.40.26`

`ssh guane`



## sinfo

- Muestra la información de los nodos y de las particiones.
- Un asterisco ( \* ) después del nombre de la partición indica que es la partición por defecto.
- Un asterisco ( \* ) después del estado del nodo indica que no esta respondiendo.

```
[user_name@guane ~]# sinfo
```

```
PARTITION   AVAIL  TIMELIMIT  NODES  STATE NODELIST
normal*     up    infinite    4     mix  guane[03,05,09,16]
normal*     up    infinite    8     alloc guane[01-02,04,10,12-15]
normal*     up    infinite    2     idle  guane[06,11]
guane_16_cores up    infinite    2     mix  guane[03,05]
guane_16_cores up    infinite    1     idle  guane06
guane_24_cores up    infinite    2     mix  guane[09,16]
guane_24_cores up    infinite    8     alloc guane[01-02,04,10,12-15]
guane_24_cores up    infinite    1     idle  guane11
Viz         up    infinite    1     idle  yaje
deepL       up    infinite    1     alloc felix
```

```
queue -u druedap
```

- Muestra la cola de trabajos del usuario *druedap*

```
JOBID      PARTITION  NAME      USER  ST      TIME  NODE  NODELIST(REASON)
18276      deepL      mafft_09_mpi  druedap  R      7:46:29  1  felix
18277      normal    gisaid_04  druedap  R      7:39:41  1  guane02
18282      guane_24_cores  gisaid_03  druedap  R      2:33:47  1  guane04
```

```
[user_name@guane ~]# queue
```

## STATUS

**R** = Running

**PD** = Pending

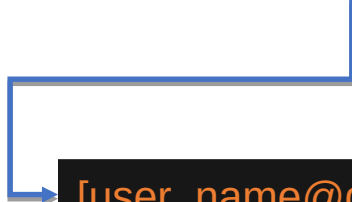
**CA** = Cancelled

```
JOBID      PARTITION  NAME      USER  ST      TIME  NODE  NODELIST(REASON)
17772      guane_24_cores  boinc     latorresn  R      23-08:13:22  1  guane10
18014      guane_24_cores  siml      ccbernalc  R      10-22:24:19  1  guane15
18015      normal      orcaLNi   geramirez  R      10-21:48:41  1  guane01
18046      guane_24_cores  siml      ccbernalc  R      9-12:22:39  1  guane13
18252      normal      cubes3.sh jmpacheco  R      22:22:32  1  guane03
18275      guane_24_cores  siml      arromerob  R      8:28:06  1  guane14
18276      deepL      mafft_09_mpi  druedap  R      7:47:35  1  felix
18277      normal    gisaid_04  druedap  R      7:40:46  1  guane02
18279      guane_24_cores  siml      arromerob  R      6:09:15  1  guane12
18281      normal      bash      emvargasd  R      4:40:48  1  guane16
18282      guane_24_cores  gisaid_03  druedap  R      2:34:52  1  guane04
18283      guane_16_cores  cubes1.sh crcarvajal  R      1:28:47  1  guane05
18284      guane_24_cores  cubes2.sh crcarvajal  R      1:26:58  1  guane09
```

## `srun options`

- Permite ejecutar una aplicación de forma directa con las opciones determinadas por el usuario en *options*

```
[user_name@guane ~]# srun --ntasks=4 --partition=normal --label /bin/hostname
```



```
[user_name@guane ~]# srun -n 4 -p normal -l /bin/hostname
```

```
2: guane01.uis.edu.co  
1: guane01.uis.edu.co  
0: guane01.uis.edu.co  
3: guane01.uis.edu.co
```

## `salloc` *options*

- Obtiene la asignación de un trabajo con acceso a consola
- Los recursos reservados para el trabajos son los especificados en *options*
- Permite realizar una reserva interactiva

## RESERVA INTERACTIVA

```
[user_name@guane ~]# salloc --nodes=1 --partition=normal --exclusive srun --pty /bin/bash
```

```
[user_name@guane ~]# srun --nodes=1 --partition=normal --exclusive --pty /bin/bash
```



## MODULOS DE AMBIENTE – SOFTWARE DISPONIBLE EN GUANE

- Los módulos son un empaquetado de variables de entorno dentro de un script.
- Se define un modulo por aplicación y este define un ambiente apropiado para su ejecución.
- **Lista de comandos:**
  - `module avail`
  - `module load MODULE_NAME`
  - `module unload`
  - `module list`
  - `module purge`

## MODULOS DE AMBIENTE – SOFTWARE DISPONIBLE EN GUANE

### module avail

- Muestra todos los módulos disponibles en la plataforma

```

----- /opt/ohpc/pub/modulefiles -----
Analytics/Anaconda/python3
Analytics/Darknet/1.0
Analytics/Julia/1.0.5
Analytics/Julia/1.2.0 (D)
Analytics/Octave/5.1.0
Bioinformatics/Bioconda/python3
Bioinformatics/Geneious/9.1.8
Bioinformatics/NGSEP/4.0.1
Bioinformatics/Spread3/0.9.6
Bioinformatics/TempEst/1.5.3
Bioinformatics/clustalOmega/1.2.4
Bioinformatics/jmodeltest/2.1.10
Bioinformatics/megaCC/10.1.8
CAE/ansys/2020r1
CFD/OpenFOAM/2.4.0
CFD/OpenFOAM/1906 (D)
Chemistry/gamess/2019R2
Chemistry/gromacs/2018.8_GPU
Chemistry/gromacs/2019.3 (D)
Chemistry/nwchem/6.8
Chemistry/orca/4.0.1.2
Chemistry/orca/4.2 (D)
EasyBuild/3.9.4
Matlab/R2020a
QuantumATK/2018.06-SP1-1/2018.06-SP1-1
QuantumATK/2019.03-SP1/2019.03-SP1
QuantumExpresso/6.5
autotools
boinc/7.14.2
clustershell/1.8.2
cmake/3.15.4
comsol/5.3a
containers/docker/19.03.9
devtools/cmake/3.14.3
devtools/cuda/7.5
devtools/cuda/8.0
devtools/cuda/9.1
devtools/cuda/10.1 (D)
devtools/gcc/5.3.0
devtools/gcc/6.2.0
devtools/gcc/7.4.0
devtools/gcc/8.3.0
devtools/gcc/9.2.0 (D)
devtools/globalarrays/5.6.1
devtools/intel/2016.4
devtools/intel/2017.8
devtools/intel/2019.4
devtools/intel/2020.1 (D)

----- /opt/ohpc/admin/spack/0.12.1/share/spack/modules/linux-centos7:
ncurses-6.1-gcc-8.3.0-fazhf5h openblas-0.3.3-gcc-8.3.0-byhg6e2 pcre-8.42-gcc-8.3.0-4rago5n pkgconf-1.4.2-gcc-8.3

```

Where:

D: Default Module

Use "module spider" to find all possible modules.

Use "module keyword key1 key2 ..." to search for all possible modules matching any of the "keys".

## MODULOS DE AMBIENTE – SOFTWARE DISPONIBLE EN GUANE

**module load** *module\_name*

- Carga las variables de ambiente correspondientes al modulo seleccionado (*module\_name*)

```
[user_name@guane ~]# module load CFD/OpenFOAM/1906
```

**module list**

- Lista todos los módulos que hallan sido cargados con el comando **module load**. Debe tener presente que puede cargar uno o más módulos de forma simultanea

## MODULOS DE AMBIENTE – SOFTWARE DISPONIBLE EN GUANE

### **module unload**

*module\_name*

Elimina todas las variables de ambiente correspondientes al modulo seleccionado (*module\_name*)

```
[user_name@guane ~]# module unload CFD/OpenFOAM/1906
```

### **module purge**

- Elimina todas las variables de ambiente de todos los módulos que se encuentren cargados en la sesión actual

## CREAR UN BATCH JOB SCRIPT

myjob.slurm

```
#!/bin/bash
```

```
# Solicitud de recursos
```

```
#SBATCH --partition=guane_16_cores
```

```
#SBATCH --nodes=1
```

```
#SBATCH --ntask=1
```

```
#SBATCH --ntasks-per-node=1
```

```
#SBATCH --mem=1G
```

```
# Tiempo de ejecución del trabajo
```

```
#SBATCH --time=1-12:30:00
```

```
# nombre del trabajo y archivos de salida
```

```
#SBATCH --job-name=myjob
```

```
#SBATCH --output=myjob.out
```

```
#SBATCH --error=myjob.err
```

```
# Carga del modulo de ambiente
```

```
module load CFD/OpenFOAM/1906
```

```
blockMesh
```

### Preliminares

- Especificar el interprete de comando (Bash)
- Siempre debe ser la primera línea

### Directivas de SLURM

- Siempre deben empezar con **#SBATCH**
  - Son ignoradas por **bash** pero interpretadas por SLURM
- Pueden realizarse comentarios antes, entre o después de las directivas
- Deben colocarse antes de la carga de los módulos y de la ejecución del trabajo

### Comandos del SCRIPT

- Carga de los módulos requeridos para la ejecución del trabajo
- Comandos que se quieren ejecutar en los nodos de computo
  - Ejecutable de la aplicación cargada
  - Pueden escribirse comandos de programación en bash

```
sbatch batch_file
```

- Envía el *batch\_file* a SLURM para su ejecución
- Si el envío es exitoso, SLURM retorna el ID del trabajo

```
[user_name@guane ~]# sbatch myjob.slurm
```

```
[user_name@guane ~]# squeue -u druedap
```

JOBID	PARTITION	NAME	USER	ST	TIME	NODE	NODELIST(REASON)
18276	deepL	mafft_09_mpi	druedap	R	7:46:29	1	felix
18277	normal	gisaid_04	druedap	R	7:39:41	1	guane02
18282	guane_24_cores	gisaid_03	druedap	R	2:33:47	1	guane04

```
scancel jobid
```

- Envía una señal al job y/o a sus subprocessos
- Por defecto la señal que se envía es SIGKILL para la terminación del job.
- El trabajo que se cancela es aquel que corresponda con *jobid*
- El *jobid* se obtiene al ejecutar el comando `sinfo`

```
[user_name@guane ~]# scancel 12345
```

- Se pueden usar filtros para la cancelación de trabajos

```
[user_name@guane ~]# scancel --user=druedap
```